Mathematics and Computers in Simulation 218 (2024) 248–265

Contents lists available at ScienceDirect

Mathematics and Computers in Simulation

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

Original articles

Gooseneck barnacle optimization algorithm: A novel nature inspired optimization theory and application

Marzia Ahmed [a,b], Mohd Herwan Sulaiman [a], Ahmad Johari Mohamad [a,*], Mostafijur Rahman [c]

[a] Faculty of Electrical & Electronics Engineering Technology, University Malaysia Pahang Al-Sultan Abdullah, Pekan, Malaysia
[b] Department of Software Engineering, Daffodil International university, Daffodil Smart City, Ashulia, Dhaka, Bangladesh
[c] Department of Computer Science and Engineering, Green University of Bangladesh, Bangladesh

ARTICLE INFO

Keywords:
Optimization
Evolutionary algorithm
Meta-heuristic
Constrained optimization
Benchmark
Covid-19 confirmed cases

ABSTRACT

This paper introduces the Gooseneck Barnacle Optimisation Algorithm (GBO) as a novel evolutionary method inspired by the natural mating behaviour of gooseneck barnacles, which involves sperm casting and self-fertilization. GBO is mathematically modelled, considering the hermaphroditic nature of these microorganisms that have thrived since the Jurassic period. In contrast to the previously published Barnacle Mating Optimizer (BMO) algorithm, GBO more accurately captures the unique static and dynamic mating behaviours specific to gooseneck barnacles. The algorithm incorporates essential factors, such as navigational sperm casting properties, food availability, food attractiveness, wind direction, and intertidal zone wave movement during mating, creating two vital optimization stages: exploration and exploitation. Real-world case studies and mathematical test functions serve as qualitative and quantitative benchmarks. The results demonstrate that GBO outperforms well-known algorithms, including the previous BMO, by effectively improving the initial random population for a given problem, converging to the global optimum, and producing significantly better optimization outcomes.

1. Introduction

Finding the optimal answer to a problem within specified limits and with a known objective function is called optimization. Given constraints, optimization seeks a solution that minimizes or maximizes an objective function. Most optimization algorithms are based on an iterative search strategy that tries to narrow down the possible outcomes until one is found that meets the criteria being optimized for. Steps in the search process include making a list of possible solutions and ranking them based on how well they meet the goal. After this assessment, the algorithm revises the pool of candidates it is considering and keeps looking for the best one until it finds it or hits a predetermined stopping point. Algorithms for optimization that take insights from the fundamentals of nature are known as "nature-based meta-heuristic optimization algorithms". These algorithms try to find the best answer possible by imitating natural processes like evolution, swarm intelligence, and natural selection. Engineering, finance, and machine learning are just a few of the areas in which these techniques are extensively utilized. The algorithm chosen depends on the problem and how the solution space is set up [17,47,49,58,68,74].

Genetic algorithms (GA) [29,31,36] are population-based search methods that mimic the principles of natural selection and genetics. They start with an initial population of potential solutions, represented as individuals or "chromosomes". The algorithm

* Corresponding author.
E-mail addresses: ahmed.marzia32@gmail.com (M. Ahmed), herwan@umpsa.edu.my (M.H. Sulaiman), johari@umpsa.edu.my (A.J. Mohamad), mostafijur@cse.green.edu (M. Rahman).

https://doi.org/10.1016/j.matcom.2023.10.006
Received 4 April 2023; Received in revised form 4 October 2023; Accepted 8 October 2023
Available online 3 November 2023

List of abbreviations

GBO	Gooseneck Barnacle Optimisation
BMO	Barnacle Mating Optimizer
GA	Genetic algorithms
PSO	Particle Swarm Optimization
ACO	Ant Colony Optimization
DE	Differential Evolution
ABC	Artificial Bee Colony
GWO	Grey Wolf Optimizer
COA	Coati Optimization Algorithm
GOA	Gazelle Optimization Algorithm
WWPA	Water Wheel Plant Algorithm
GLA	Group Learning Algorithm
FHO	Fire Hawk Optimizer
NFL	No Free Lunch
FDO	Fitness Dependent Optimizer
DA	Dragonfly Algorithm
WOA	Whale Optimization Algorithm
SSA	Salp-Swarm optimization Algorithm
TF	Test Function
TCV-W1D1	Total Cumulative Vaccination-Week1Day1
MAPE	Mean Absolute Percentage Error
MVO	Multi-Verse Optimizer
EMA	Evolutionary Mating Algorithm
LSSVM	Least Square Support Vector Machine
HBA	Honey Badge Algorithm

then applies selection, crossover, and mutation operations to create future candidate solutions. The selection process favours individuals with better fitness values, resembling the survival of the fittest. Crossover combines genetic information from two parent solutions to create offspring, while mutation introduces random changes in the population to encourage exploration. The process continues until a satisfactory solution is found. Particle Swarm Optimization (PSO) is a swarm intelligence-based algorithm inspired by the coordinated movement of bird flocks or fish schools. In PSO, each potential solution is represented as a "particle" that moves through the solution space. The particles adjust their positions and velocities based on their experience and the swarm's best-known position. This cooperative behaviour encourages particles to move towards promising regions of the solution space, eventually converging to an optimal solution [23,46,66]. The efficient foraging behaviour of ant colonies inspires the algorithm named Ant Colony Optimization (ACO) [21]. In this algorithm, artificial ants traverse the solution space, leaving pheromone trails as they explore. The amount of pheromone on a path is updated based on the solution quality found along that path. Ants are more likely to follow paths with higher pheromone levels, thus reinforcing the exploration of promising regions. Over time, ACO converges on the optimal solution as ants prefer paths with more robust pheromone trails. Differential Evolution (DE) [75,85] is a population-based optimization method that maintains a population of candidate solutions. New solutions are generated by adding the scaled difference between randomly chosen individuals to a target individual. The new solutions are then evaluated, and if they offer better performance, they replace the corresponding target individual. DE's mutation and selection processes effectively balance exploration and exploitation, leading to the discovery of optimal or near-optimal solutions. ABC, or Artificial Bee Colony [39,81], draws inspiration from the foraging behaviour of honeybees. In this algorithm, bees represent potential solutions, searching for better solutions by exploiting known sources and exploring new ones. The employed bees exploit known solutions, while onlooker bees assess and select new solutions based on their quality. Additionally, scout bees explore uncharted areas to discover potentially better solutions. This combination of exploitation and exploration enables ABC to search the solution space efficiently. Grey Wolf Optimizer (GWO) [54] takes inspiration from the hierarchical hunting behaviour of grey wolves. The algorithm organizes a pack of wolves, each representing a potential solution, into an alpha, beta, delta, and omega hierarchy. The alpha wolf represents the best solution found so far. During the iterative process, wolves move towards the positions of higher-ranking wolves, guided by their dominance in the hierarchy. The hierarchy is dynamically adjusted as new solutions are found. This hierarchical structure and movement ensure practical exploration and exploitation in the search for the optimal solution. The Coati Optimization Algorithm (COA) [20] is a newly developed bioinspired optimization algorithm inspired by the foraging behaviours of coatis in nature. By modelling the coatis' strategies when attacking iguanas and escaping from predators, COA draws fundamental inspirations for its design. The algorithm's implementation consists of two phases: exploration and exploitation, which are mathematically modelled based on hunting and escape strategy simulations. The Gazelle Optimization Algorithm (GOA) [4] is a novel metaheuristic algorithm inspired by the survival abilities of gazelles in their predator-dominated environment. The algorithm is designed to tackle real-world optimization problems by leveraging the gazelles' strategies for evading predators. The GOA is modelled as a population-based metaheuristic algorithm with a stochastically generated initial population. The optimization process consists of two phases: the exploitation phase, which simulates grazing peacefully or being stalked by a predator, and the exploration phase, where gazelles outmaneuver and outrun the predator to reach a haven. Different equations mathematically model each phase, and the algorithm repeats these phases subject to termination criteria to find optimal solutions to optimization problems. The Fire Hawk Optimizer (FHO) [10] is a novel metaheuristic algorithm inspired by the foraging behaviour of Fire Hawks, which include whistling kites, black kites, and brown falcons. The algorithm's design is based on these birds' specific actions, including setting fires to catch prey in nature.

Barnacle Mating Optimizer (BMO) [55–57,76] is another metaheuristic optimization algorithm named after barnacle mating rituals. BMO mimics barnacle mating, which releases sperm and eggs into the ocean and produces offspring. In the first stage, Candidate solutions were generated using a random generator. The mathematical model, including initialization, parent selection, and offspring creation, has been reported in [65,71]. To produce new offspring, mating is done randomly. The tuning parameter penis length (pl) represents the variety of barnacles that may mate. Because they can both generate and take sperm, barnacles are thought to be hermaphrodites. It is thought that each barnacle can only be fertilized by one barnacle at a time to simplify the algorithm. Self-mating can happen in nature, although it seldom does. In BMO, barnacle locations represent optimization solutions. The algorithm selects a group of male and female barnacles with the most significant and lowest fitness ratings at the start of each breeding cycle. Then, it merges their positions to generate a new generation. The best offspring replace the weakest barnacles and some processed repeat until a solution is found. What makes BMO easy to use is that you do not have to fine-tune many parameters. The algorithm is also flexible in that it can be used to solve both continuous and discrete problems in the optimization space.

Even though the results look promising, BMOs after several improvement done by different researchers [22,24,26,40,83] use of real-world case studies still needs to be revised with more questions about how stable, scalable, and fast the algorithm is. Additionally, more theoretical analysis is required to completely appreciate BMO's activities and consequences. Implementing the original BMO exemplifies the characteristics of pseudo-copulation mating since it bears little resemblance to BMO's reproduction or mating behaviour. In the original BMO, the real-life scenario has yet to be considered appropriately. It only took one property of BMO, the penis length, and the algorithm was made based on the pseudo-copulated mating without considering any constraint. Moreover, for the complex dataset, BMO could have performed better. One of the species named is the gooseneck barnacle, which releases sperm into the water, known as sperm casting mating behaviour, followed by the maximum barnacles for reproduction. These constraints must be considered by the BMO algorithm. Also, the initialization method impacts the performance of the Barnacle Mating Optimization (BMO) algorithm. If the initial population gives more than one solution, the optimization process may need to be able to search the search space. If the BMO algorithm gets trapped in a local optimum, it may be unable to discover the global optimum and provide accurate results. The algorithm tries to balance BMO's exploration and exploitation stages using a parameter that does not work well (penis length).

Even if there have been a lot of recent papers [25,30,37,50,53,61,62,78–80] in this area, other creatures with unique behaviour in nature continue to be noticed. For example, gooseneck barnacles are a different kind of barnacle that is an upscale and expensive culinary item like lobster. This work will first try to figure out the essential parts of gooseneck barnacles since no study has tried to imitate their natural ability to live and reproduce. Next, depending on the noted traits, an algorithm is suggested. The reason for this work's proposal of this optimizer is also supported by the "no free lunch" (NFL) [82] theorem since this approach may perform better than others on specific problems that have not yet been solved.

Ultimately, the following are the most important results from this study:

- First, we present a novel population-based optimization technique called GBO, which is modelled after the reproductive and ecological processes of the gooseneck barnacle.
- Second, it compares GBO to the best available metaheuristic algorithms regarding its statistical significance, convergence speed, exploitation-exploration ratio, and diversity.
- Third, we conduct a series of tests to see how the proposed algorithm performs compared to other solutions to benchmark optimization problems, the CEC'2019 simulation, and the real-world application and challenges.
- Finally, in challenging optimization issues, GBO surpasses its rivals.

The remaining sections are arranged as follows:

The motivation and biological underpinnings of the study are presented in Section 2. In Section 3, the mathematical models and the GBO algorithm are presented. To verify and validate the performances of GBO, Section 4 provides a thorough comparison analysis on several benchmark functions and one actual case study. Section 5 concludes and offers some suggestions for further research.

2. Gooseneck barnacle optimization algorithm

2.1. Inspiration

The microorganisms known as barnacles have been around since the Jurassic period. Barnacles are born with the ability to swim. As they age, they stick to things in the water and grow shells. Most barnacles are hermaphrodites, meaning they can reproduce both male and female. Barnacles are abundant in the world's waterways; acorn barnacles are the most prevalent. Barnacles have penises

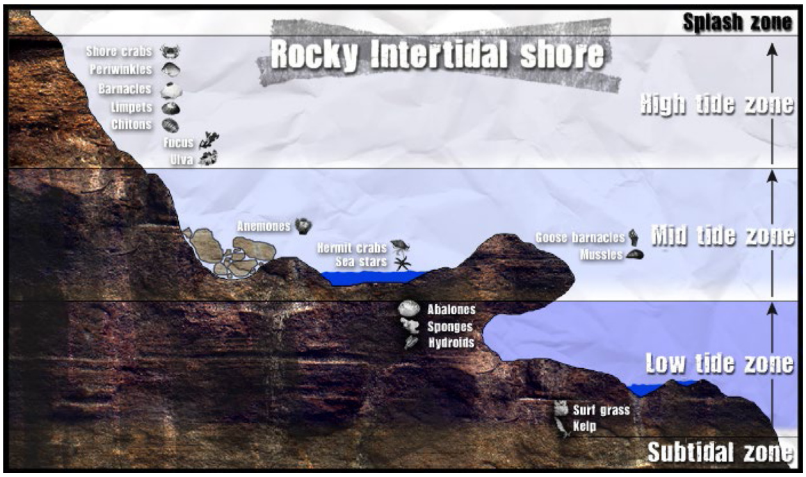

Fig. 1. The intertidal zone where gooseneck barnacle lives [38].

that are seven to eight times the length of their bodies, which helps them adapt to their aquatic environment and stationary lifestyle while also making them a fascinating study subject [11]. Mating groups are formed by aggregating individuals in specific areas on substrates, driven by chemical cues and physical interactions. Barnacles are hermaphrodites, meaning they have both male and female reproductive organs. During mating, the pennis of neighbouring barnacles come into contact, and they use these appendages to transfer sperm to one another. Sperm transfer occurs during mating encounters, where mating groups are formed, and individual barnacles compete for mate access. This competitive behaviour is crucial in determining reproductive success. Barnacles form mating groups with their immediate neighbours, including possible mates and potential rivals for mating. Regarding mating group size and local mate competition, individual variation in penis reach may have a significant effect [32,42]. This study pointed out some problems with BMO that significantly affect how well the original algorithm works and fixes them. Unfortunately, the initial design should have considered several crucial aspects of BMO. While many other forms of barnacles exist, the gooseneck barnacle is the one we will discuss in this article.

Gooseneck barnacles, also known as stalked barnacles, are a type of crustacean that lives attached to hard surfaces in the ocean, such as rocks, shells, and other animals. They are called gooseneck due to their long, flexible, and curved stalks that attach them to surfaces barnacles shows in Fig: 1 [38] and 2 [63]. Gooseneck barnacles reproduce sexually. The males and females release their gametes into the water to fertilize each other's eggs. The fertilized eggs hatch into larvae that drift in the water until they find a suitable place to settle and grow [3]. The advantage of the gooseneck barnacle's mating procedure is that it allows them to reproduce in high-density populations. In addition, because they release their gametes into the water column, fertilization can occur even when individuals are physically separated, which can be helpful in crowded areas with little space to move and mate (see Fig. 1).

Researchers from the University of Alberta in Edmonton and the Bamfield Marine Sciences Centre in British Columbia looked at the gooseneck barnacle (*Pollicipes polymerus*). They found a new way for it to reproduce. This disproved a theory that had been around for more than 150 years. The scientists had previously noted that self-fertilization had never been noted in earlier examinations of the gooseneck barnacle. They also saw that the barnacles in the field were leaking sperm, which led them to wonder whether they may have picked up sperm from the water. To do the genetic analysis of the paternal combinations, the researchers took gooseneck barnacles from Barkley Sound in British Columbia, alone and in pairs, along with their fertilized eggs. One hundred percent of these eggs must have been fertilized by catching sperm from the sea, since the DNA of the fertilized eggs showed that none of the separated barnacles had created embryos by self-fertilization [3,11,32,38,42,63,67].

When it comes to barnacle penises, length is not the only factor. Longer penises might help attract more potential mates, but they are more likely to flail around in big waves. The water conditions where barnacles live can change the size and shape of their penises in a big way. This helps them fight waves in rough seas. Barnacles in calm seas have long, flexible penises to reach as many possible mates as possible. However, the penises of barnacles inhabiting rougher, more turbulent seas are shorter and thicker. Giant penises cannot reach as far as smaller ones, but because they have more support, they are better able to deal with waves that get in the way. They are more likely to successfully reach their neighbours because they are more robust, less prone to breaking in the surf, and less likely to be tossed about by the waves. When barnacles move from calm seas to seas with more waves, their penises will change shape proportionally. Before breeding season, barnacles regenerate their penises every year. No matter what kind of penis its parents have or what kind of penis it had the last time it mated, a barnacle in choppy water will grow a short, strong one. Fig. 3 shows the lifecycle of an adult gooseneck barnacle.

To mate with faraway neighbours, the barnacles release their sperm into the ocean, known as "sperm casting". Then, barnacles that look like women catch the sperm and use it to make their eggs grow. Fig. 4 explains the evolution happens in every phase of

Fig. 2. Gooseneck barnacles [63].

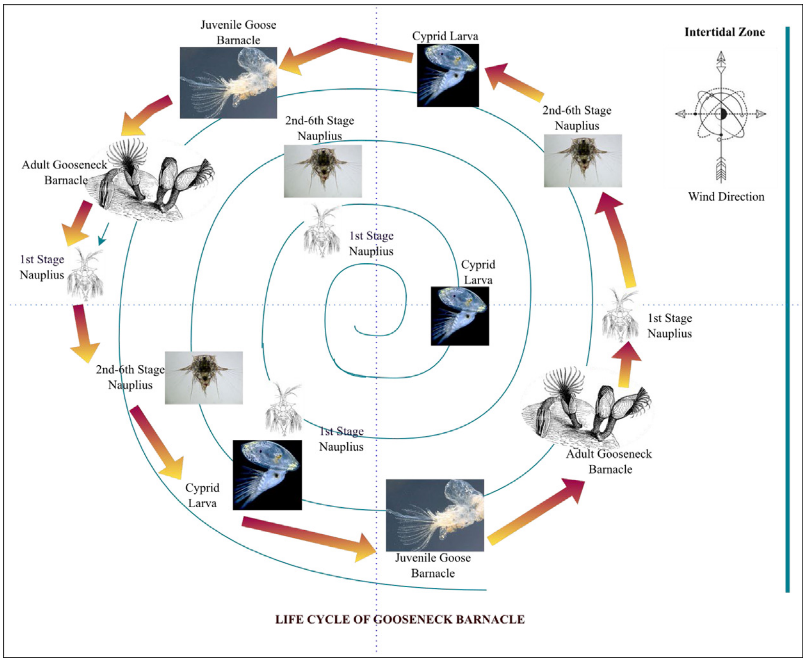

Fig. 3. Life cycle of gooseneck barnacle or stalked barnacle.

gooseneck barnacle in their life cycle. Several marine animals, such as starfish, dog winkles, and ribbon worms, prey on barnacles because they provide a food source for plankton. It feeds by sweeping the water with its legs, or cirri, to gather food and expel waste. Moreover, the wave action (height and velocity or intensity) affects the reproduction of gooseneck barnacles [13].

Sperm casting is not the same as broadcast spawning, which is when both males and females put their eggs and sperm into the water at the same time. Other animals that use sperm casting include sponges, corals, and molluscs. These barnacles are the first crustaceans known to use this strategy. For an animal that cannot move, the barnacle has found strange ways to find and reach mates nearby and far away [12,33–35,41,59]. This behaviour has been observed and identified through the included articles and modelled mathematically to propose an optimizer called Gooseneck Barnacle Optimizer (GBO) in the following section.

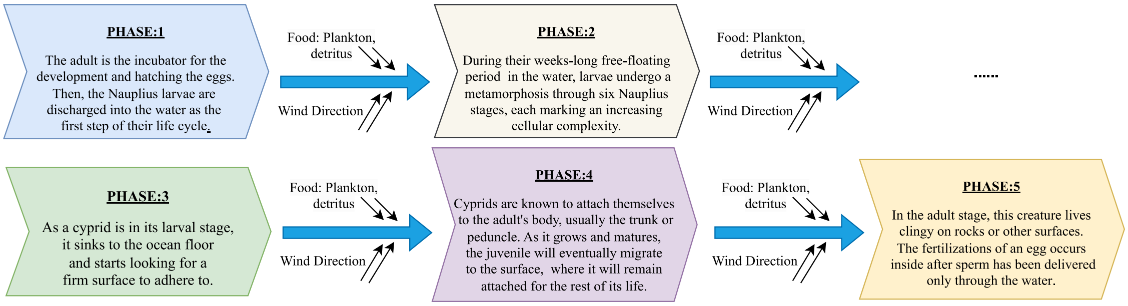

Fig. 4. Phases of lifecycle.

2.2. GBO algorithm

The most important characteristic of gooseneck is the mating behaviour through sperm casting and how the wave action and wind affect the reproduction. Therefore, if we find a way to mathematically model these characteristics, we can design an algorithm.

The mathematical model can be designed as follows:

$$(X+l)_{i+1} = (X+l)_i + WD_i + T_{dim} + S\left((X+l)_{i,}\left(Sp_{water}\right)_j\right) + Hs.(X+l)_i \tag{1}$$

where, $(X+l)$ defines the position of goose barnacle in ith iteration, WD is the direction of wind in ith iteration, T_{dim} is the target dimension to move towards the target or to the best solution, $S\left((X+l)_{i,}\left(Sp_{water}\right)_j\right)$, S is the logarithmic spiral between the jth sperm region in water, Sp_{water} and ith $(X+l)_i$, Hs is the significant wave height. The detailed explanation of each phase of the mathematical model has been discussed in the following section.

2.2.1. Initialization

In the proposed GBO algorithm, it is assumed that the candidate solution are gooseneck barnacles and the variables for the problem space are the position of gooseneck. The gooseneck can travel any dimensional space with extending their penis, that can be seven/eight times extendable to their body. Since GBO is a population-based algorithm, for initialization, the candidate of solution X is the population of gooseneck barnacles is a type of stalked barnacle, which can be represented as follows:

$$X = \begin{bmatrix} (x+l)_{1,1} & (x+l)_{1,2}\cdots & (x+l)_{1,d} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ (x+l)_{n,1} & (x+l)_{n,2}\ldots & (x+l)_{n,d} \end{bmatrix} \tag{2}$$

where n and d are the total populations and the number of dimension or variables to be optimized. Since the gooseneck has an edible structure [27,41] (see figure Fig. 2), each has varied length, l. which will be taken randomly. For all goosenecks, there will be a corresponding region in the water which contains sperm for mating. This is another key component in the proposed algorithm named as 'sperm_region' which has similar matrix as X as follows:

$$\text{Sperm\_Region} = \begin{bmatrix} (sr)_{1,1} & (sr)_{1,2}\cdots & (sr)_{1,d} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ (sr)_{n,1} & (sr)_{n,2}\ldots & (sr)_{n,d} \end{bmatrix} \tag{3}$$

where n is the number of gooseneck and d is the number of variables or dimension as X. While the location of gooseneck barnacles and sperm areas are both possible answers, the way in which the problem domain handles them in each iteration will vary. Real search agents, like those found in the penis of gooseneck barnacles, navigate the search space, while the sperm area in the water represents the optimal mating location. For the sake of mating and reproduction, the sperm area may be thought of as the new gooseneck in the search space. Thus, the goosenecks do a search radius around the new position and update it if they discover a better alternative. This method ensures that the optimal option for a gooseneck will always be available.

Some studies [15,27,45] prove that Gooseneck barnacles are found in the upper and intermediate intertidal zones. Significant wave height is considerably below 0.8 to 1.5–3 m above mean low water is the dominance for barnacles [15,45]. Generally, significant wave height is calculated using this equation $Hs = 4\sqrt{H\hat{\ }2T}$ [70–73], however, in this version of GBO we are going to calculate Hs, using following equation (considering the tolerance of barnacles in their life cycle):

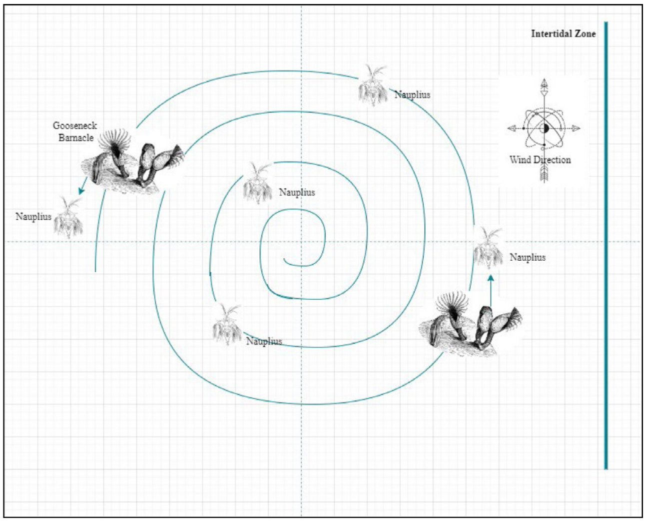

Fig. 5. Logarithmic spiral around the gooseneck in intertidal zone of a sea.

Significant Wave Height,

$$Hs = 1.5 - \left(\frac{Iteration(1.5 - 0.2)}{Maximum\ Iteration}\right) \tag{4}$$

where, through literature it has been found that barnacle can tolerate the wave intensity between 0.2–1.5 or 3.0 m, Hs will be decreasing in every iteration to explore the range of wave intensity to find out the optimal mating region.

For sperm casting, a neighbouring logarithmic spiral region is defined, as follows:

$$S\left((X+l)_{i,}\left(Sp_{water}\right)_j\right) = D_i.e^{bt}.\cos(2\pi t) + \left(Sp_{water}\right)_j \tag{5}$$

where Di indicates the distance of the ith barnacle and for the jth sperm region for mating, b is a constant for defining the shape of the logarithmic spiral, and t is a random number in [−1, 1]. Distance is calculated as follows: $D_i = (X+l)_{i,} - \left(Sp_{water}\right)_j$. where $(X+l)_{i,}$ indicate the ith barnacles, $\left(Sp_{water}\right)_j$ indicates the jth sperm casting region, and Di indicates the distance of the ith barnacles for the jth sperm casting region. The logarithmic spiral [25,30,48] has been considered as shown in Fig. 5, considering the real scenarios.

2.2.2. Off-spring generation

The behaviour of sperm-cast mating is assumed to be the combination of a few patterns discussed in this paper. To update the position of new gooseneck barnacles in a search space and simulate the movement towards new offspring generation we have considered a movement vector, Δ(X + l)i defined as follows:

$$\Delta(\mathrm{X}+\mathrm{l})_{\mathrm{i}+1} = \mathrm{WD_i} + \mathrm{T_{dim}} + \mathrm{Hs}.\Delta(\mathrm{X}+\mathrm{l})_{\mathrm{i}} \tag{6}$$

$$(\mathrm{X}+\mathrm{l})_{\mathrm{i}+1} = (\mathrm{X}+\mathrm{l})_{\mathrm{i}} + \Delta(\mathrm{X}+\mathrm{l})_{\mathrm{i}+1} \tag{7}$$

$$(\mathrm{X}+\mathrm{l})_{\mathrm{i}+1} = (\mathrm{X}+\mathrm{l})_{\mathrm{i}} + \mathrm{Levy} * (\mathrm{X}+\mathrm{l})_{\mathrm{i}} \tag{8}$$

WD_i defines the wind direction which has been derived through the degree range [0 359] on the radius of the search space, and finally added with the target dimension $\mathrm{T_{dim}}$ to the best solution. Dimension towards target, T_{dim} is the value of the dimension in the target, by assuming wind direction always to the target.

Fig. 6 shows the exploration process of GBO.

where *Levy* flight [45] is determined as follows:

$$Levy(N) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|^{\frac{1}{\beta}}} \tag{9}$$

β is a constant set to 1.5, r1 and r2 are random numbers [0-1], and the equation is as follows:

$$\sigma = \left(\frac{\tau(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\tau\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \tag{10}$$

where $\tau(y) = (y-1)!$

The previous explanation shows that the suggested mathematical model calls for the gooseneck barnacle to move toward a target mating zone in the logarithmic spiral sperm region for sperm casting over several iterations. However, there are no directions in the actual search space because we still need to find out where the global optimum is. Hence, in each optimization iteration, we must choose a mating goal. In GBA, the gooseneck with the highest objective value at the time of optimization is thought to be the goal, which forces other goosenecks to move closer to the mating zone. However, it also helps GBA save the most promising target in the search space during each repetition. The GBA algorithm's pseudocode is given in the next section. The GBA begins the optimization process by generating a random pool of candidate solutions. Then, based on Eq. (7), the search agents change their positions towards the target considering the wind direction and significant wave activity or action. For mimicking the real scenario sometime there may exists only one sperm region left, due to the wave intensity and wind. In that case, their position will be updated using Eq. (8). Finally, iteratively updating the position is done until the final requirement is met. In the end, the best global estimate is given, along with the location and quality of the best target. While the previous simulations and discussions illustrated how the GBA algorithm is successful at locating the best solution in a search space.

2.2.3. Pseudo code of Gooseneck barnacle

Initialize the Gooseneck barnacle population (search agents) $(X+l)_i$ (i = 1, 2, ..., n) and l random length in [0,1]

while *the end criteria do not meet*
Set significant wave height, Hs
Bring the current Gooseneck barnacles back if it goes out of range or search space.
Calculate fitness of each Gooseneck barnacle.

Sort and update the position of gooseneck according to the best solution, ***GlobalBest****.*

// Selection
Allocate the mating region in the water for sperm casting eqn (2)
Sort and update the mating position for sperm casting according to the best solution, ***GlobalBest****.*

// Reproduction
for each variable
  if *Allocated sperm regions are available for corresponding gooseneck*
  // Sperm Casting (Stalked Barnacle: Gooseneck Barnacle)
    // update movement towards target
    Calculate movement vectors using Eq. (6)
    Update position of new off spring using Eq. (7)
  end

  else if *Only one Allocated sperm region is available*
  // pseudo copulation or self-fertilization (Acorn Barnacle: Barnacle)

    Update position of new off spring using Eq. (8)

  end
end for
Sort and update the best solution, ***GlobalBestMatingPosition.***
Check and correct the new positions of gooseneck and mating position for sperm casting based on the boundaries of variables.

end while

2.2.4. Computational complexity analysis

One of the most important ways to judge how long it takes for an algorithm to complete a task is by looking at its computation complexity, which varies depending on the method's design and implementation. For example, the computing cost of the GBO is a function of many parameters that define the problem, including the total number of goosenecks barnacle created during initialization, the total number of variables, the maximum number of possible iterations, and the sorting process of fitness of sperm location at each iteration. The overall complexity defined in Eq. (11)

$$O(GBO) = O(initialization\ method\ and\ boundaries) + O(fitness\ calculation\ and\ sorting) + O(New\ offSpring\ generation) \tag{11}$$

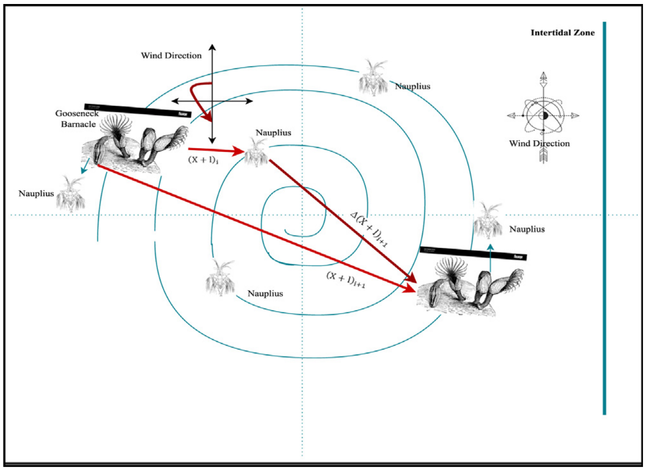

Fig. 6. Exploration process of GBO.

In concrete terms, the overall computational complexity of GBO satisfying the termination criteria can be computed as follows:

$$O(GBO) = O(n \times d) + O(t(nlogn)) + O(t(n \times d)) \tag{12}$$

$$O(GBO) = O(nd + tnlogn + tnd) \tag{13}$$

where n is the number of gooseneck barnacles, t is the maximum iteration, d is the variables defined by problem dimensions. As GBO has a polynomial-order computational complexity, it is a powerful metaheuristic optimization strategy [14]. A polynomial-order complexity is desirable as it ensures the algorithm's running time scales efficiently with the problem size, making it more practical for large-scale optimization tasks. GBO's polynomial-time complexity contributes to its power as a metaheuristic optimization strategy by enabling it to handle complex problems with larger dimensions effectively. The scalability of GBO ensures its adaptability to various real-world applications. Demonstrating polynomial-time complexity highlights GBO's competitiveness among existing algorithms and emphasizes its potential as a practical and robust optimization tool across different domains.

Some considerations for understanding the theoretical efficacy of the GBO method in addressing optimization issues are: The technique for primarily encouraging exploration and exploitation is the procedure of updating locations, which enables getting neighbouring solutions around the logarithmic spiral. Since GBO employs a community of goosenecks, it is pretty good at avoiding local optimum solutions. The likelihood of a search becoming stuck at a local optimum is reduced. The search space is explored more thoroughly when sperm are assigned to different locations around the spiral for each mating. The most viable solutions developed as of late may serve as future reference points for the gooseneck in its quest for a suitable mating area by acting as placeholders for the placements of the offspring. The performance of the suggested method is examined by using a set of mathematical functions and one difficult real-world situation in the following sections.

3. Simulation studies

Stochastic optimization often uses a collection of mathematical test functions for which the optimal values have already been determined. It is possible to compare the efficacy of various algorithms quantitatively. Still, if you want to come to a reasonable conclusion, you need to use test functions with different properties. In this part, many simulated experiments are conducted to show how successful the suggested gooseneck barnacle GBO is. The experimental results are then provided and analysed in detail.

3.1. Analysis on classical benchmark functions

Twenty-three benchmark functions were simulated to validate the searching procedure and observe the behaviour of the proposed GBO as it located the best solution. The results of GBO for the specified categories of test functions are shown in Fig. 7. This picture shows that there is only one global optimum solution for the unimodal test functions (F1 and F3). This makes them perfect for figuring out how well the GBO can be used. Still, multi-modal (F9 and F10) and composite (F15 and F18) test functions, which are meant to have a lot of local optima, have been used to test how well GBO works to avoid local optima and how well it can explore. The example uses a maximum of 200 iterations, which are shown in 2D search space.

3.2. Comparison on classical benchmark function with other metaheuristic algorithms

The test functions were solved using 30 search agents and 500 iterations. Each test function was executed 30 times to provide statistical results. After that, we used several additional metrics to statistically compare the algorithms, such as the mean and

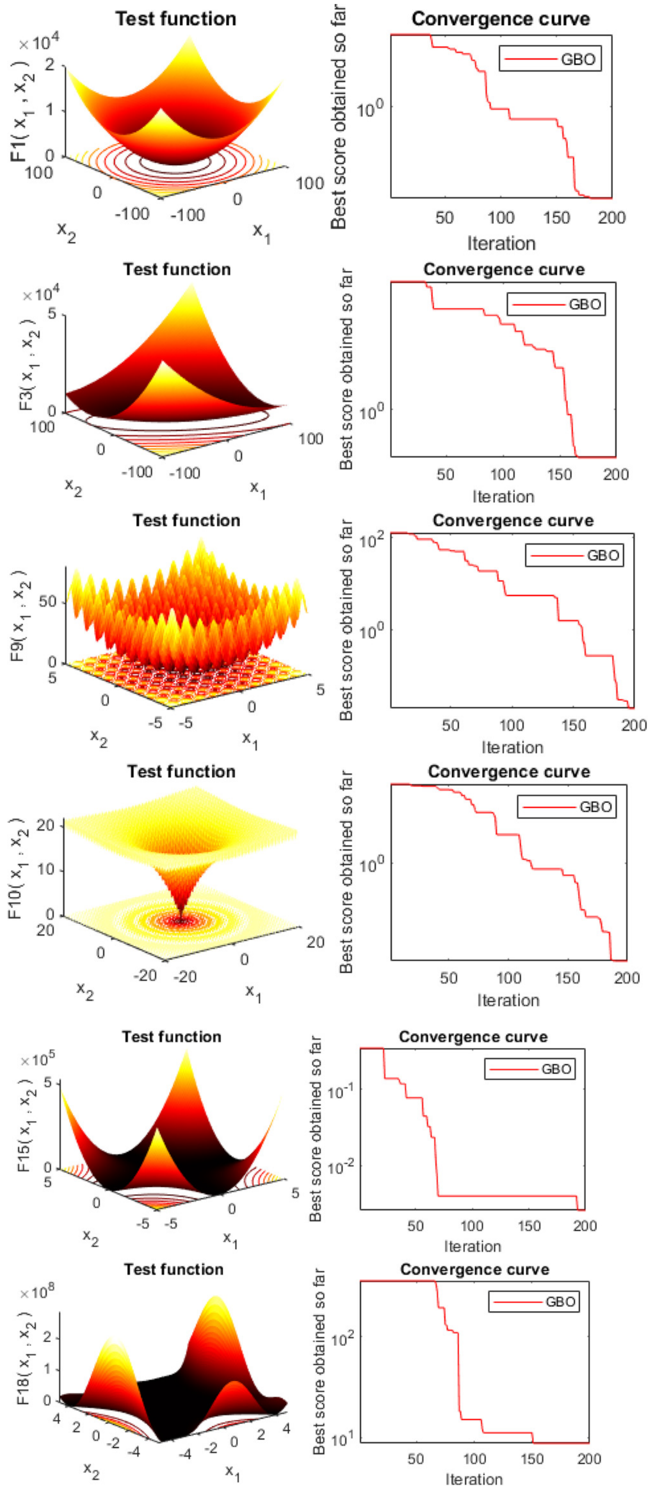

Fig. 7. Performance of GBO in selected test functions.

standard deviation of the best results. As the average and standard deviation values go down, the algorithm will be better able to avoid local solutions and find the best global solution. The findings were validated using the tools in the Table 1: PSO, BMO-Levy as IBMO, FDO (Fitness Dependent Optimizer), GA, and DA. TF 1–19 in Table 1 represents the Test Function, F1–F19.

Table 1
Classical benchmark results of selected algorithms with GBO.

Test function	GBO		IBMO		FDO		DA		PSO		GA	
	Average	STD	Average	STD	Average	STD	Average	STD	Average	STD	Average	STD
TF1	1.67E−11	5.80E−11	**2.21E−35**	**5.05E−35**	7.47E−21	7.26E−19	2.85E−18	7.16E−18	4.2E−18	1.31E−17	7.49E+02	3.25E+02
TF2	6.25E−07	1.68E−06	**2.35E−08**	**2.17E−08**	9.39E−06	6.91E−06	1.49E−05	3.76E−05	3.15E−03	9.81E−03	5.97E+00	1.53E+00
TF3	**8.40E−10**	**3.09E−09**	1.24E−01	3.97E−01	8.55E−07	4.40E−06	1.29E−06	2.10E−06	1.89E−03	3.31E−03	1.95E+03	9.94E+02
TF4	**4.04E−06**	**1.16E−05**	4.20E+00	5.92E+00	6.69E−04	2.49E−03	9.88E−04	2.78E−03	1.75E−03	0.002515	2.12E+01	2.61E+00
TF5	**8.78E+00**	**1.09E−01**	7.55E+01	1.20E+02	2.35E+01	5.98E+01	7.06E+01	6.79E+00	6.35E+01	80.12726	1.33E+05	8.50E+04
TF6	1.84E+00	1.40E−01	1.84E−12	2.36E−12	**1.42E−18**	4.75E−18	4.17E−16	1.32E−15	4.36E−17	1.38E−16	5.64E+02	2.30E+02
TF7	**3.90E−04**	**3.36E−04**	8.76E−03	3.29E−03	5.44E−01	3.15E−01	1.03E−02	4.69E−03	5.97E−03	3.58E−03	1.67E−01	0.072571
TF8	**−1.92E+03**	**1.88E−02**	−2.87E+03	3.90E+02	−2.29E+06	2.07E+05	−2.86E+03	3.84E+02	−7.1E+11	1.20E+12	−3.41E−03	1.64E+02
TF9	**6.20E−11**	**3.28E−10**	3.21E+01	1.65E+01	1.46E+01	5.20E+00	1.60E+01	9.48E+00	1.04E+01	7.88E+00	25.51886	6.67E+00
TF10	**6.57E−07**	**1.62E−06**	3.50E+00	6.96E+00	**4.00E−15**	**6.38E−16**	2.31E−01	4.87E−01	2.80E−01	6.02E−01	9.50E+00	1.27E+00
TF11	**1.32E−11**	**2.63E−11**	1.82E−01	1.01E−01	5.69E−01	1.04E−01	1.93E−01	7.35E−02	8.35E−02	3.51E−02	7.72E+00	3.63E+00
TF12	8.45E−01	2.22E−01	8.39E−01	1.78E+00	1.98E+01	2.64E+01	3.11E−02	9.83E−02	**8.57E−11**	**2.71E−10**	1.86E+03	5.82E+03
TF13	8.12E−01	4.99E−02	2.41E−02	1.13E−01	1.03E+01	7.42E+00	2.20E−03	4.63E−03	**2.20E−03**	**4.63E−03**	6.80E+04	8.77E+04
TF14	**2.08E−00**	**1.04E−00**	4.57E+00	2.99E+00	**3.79E−07**	**6.32E−07**	1.04E+02	9.12E+01	1.50E+02	1.35E+02	130.0991	2.13E+01
TF15	7.08E−03	5.91E−03	**2.18E−03**	4.95E−03	**1.50E−03**	1.24E−03	1.93E+02	8.06E+01	1.88E+02	1.57E+02	1.16E+02	1.92E+01
TF16	−6.90E−01	4.04E−01	−1.03E+00	0.00E+00	**6.38E−03**	**1.06E−02**	4.58E+02	1.65E+02	2.63E+02	1.87E+02	3.84E+02	3.66E+01
TF17	5.45E−01	3.23E−01	**3.98E−01**	**1.13E−16**	2.38E+01	2.15E−01	5.97E+02	1.71E+02	4.67E+02	1.81E+02	5.03E+02	3.58E+01
TF18	9.45E+00	9.65E+00	**3.00E+00**	**4.81E−15**	2.23E+02	9.96E−06	2.30E+02	1.85E+02	1.36E+02	1.60E+02	1.18E+02	51.00.183
TF19	−3.79E+00	7.43E−02	**−3.86E+00**	**2.71E−15**	2.28E+01	1.04E−02	6.80E+02	1.99E+02	741.6341	2.07E+02	5.44E+02	13.30161

The effectiveness of the suggested GBO has been tested on test functions from F1 to F19 and compared to those of the other methods. We simulated the classic benchmark functions; the results are shown in the table below. Thirty simulation runs have been recorded for the unimodal, multimodal, and composite benchmark functions. The FDO, GA, PSO, and DA parameter sets are all described in this paper [1], among others. Except for TF8, which was minimized to a value of −418, all other test functions were minimized to zero. For instance, multiple test functions were relocated away from it to prove that the algorithms did not favour the initial location. Unimodal benchmark functions [51], multimodal benchmark functions [51], and composite benchmark functions [51] all show the finer points of the justification and tuning of the classical benchmark functions that have been put into practice.

The results in Table 1, reveal that GBO exhibits strong performance across all these categories, showcasing its versatility and effectiveness as a metaheuristic optimization strategy. GBO demonstrates impressive convergence capabilities on unimodal functions, efficiently converging to the global optimum with high accuracy for maximum benchmarks. This indicates the algorithm's ability to exploit the search space effectively and locate the best solution within a short number of iterations.

On multi-modal functions, GBO excels in exploration, effectively exploring the search space and locating multiple promising regions, each potentially containing local optima. By leveraging its unique design inspired by the natural mating behaviour of gooseneck barnacles, GBO can effectively escape local optima and continue exploring diverse regions to find the global optimum.

Furthermore, GBO's performance on composite functions, which often represent real-world scenarios with complex and interconnected sub-functions, demonstrates its ability to handle intricate optimization landscapes. The algorithm's combination of exploration and exploitation allows it to navigate the complex search space and identify high-quality solutions in composite functions. Also, a two-sided rank sum test of the hypothesis consisting of two independent samples was performed between compared algorithms in the following sections.

3.3. Comparison on CEC-C06 2019 benchmark test functions

GBO is measured against the state-of-the-art CEC benchmark test suite of ten functions. Professor Suganthan and his team have improved these evaluating capabilities [65]. An annual tournament is held using tests taken from the 100-Digit Challenge. CEC04–CEC10 was created as a 10-dimensional minimization problem with a boundary range of [−100, 100] by the individual who created the CEC benchmark. According to [1], CEC01–CEC03 varied in size. As a result, the CEC04–CEC10 test functions are rotated and shifted, but the CEC01–CEC03 test functions are not. Scaling is an option for all tests. New 2019 CEC-C06 Standards the CEC function competition developer's disclosed function details are described in "The 100-Digit Challenge: [1]". For quality assurance, only the CEC01 findings have been changed to account for the extra 1000 dimensions applied to a subset of the functions.

Each algorithm has 500 iterations and 30 agents, so this work pits GBO against BMO-Levy, FDO, DA, WOA, and SSA regarding stated CEC functions. These algorithms have a strong reputation in the academic community and have proven successful in the laboratory and practical settings. In addition, writers often provide the code they use to construct algorithms. The competitors maintained the default values for the algorithm parameters found in their respective published works [1,51,52] throughout the testing (see Table 2).

Table 2
CEC 2019 results of selected algorithms with GBO.

IEEE ECE 2019 benchmark results												
Test function	GBO		IBMO		FDO		DA		WOA		SSA	
	Average	STD	Average	STD	Average	STD	Average	STD	Average	STD	Average	STD
CEC01	1.03E+05	3.22E+04	1.78E+10	1.30E+10	4.59E+03	2.07E+04	5.43E+10	6.69E+10	4.11E+10	5.42E+10	6.05E+10	4.75E+09
CEC02	1.86E+01	2.43E−01	1.73E+01	0.00E+00	4.00E+02	3.22E−09	7.80E+01	8.78E+01	1.73E+01	4.50E−03	1.83E+01	5.00E−04
CEC03	1.27E+01	2.59E−04	1.27E+01	1.25E−08	1.37E+03	1.65E−11	1.37E+01	7.00E−04	1.37E+03	0.00E+00	1.37E+01	3.00E−04
CEC04	1.45E+04	3.87E+03	9.60E+01	1.06E+02	3.41E+01	1.65E+01	3.44E+02	4.14E+02	3.95E+02	2.49E+02	4.17E+01	2.22E+01
CEC05	4.73E+00	5.68E−01	1.29E+00	1.95E−01	2.14E+00	8.58E−02	2.56E+00	3.25E−01	2.73E+00	2.92E−01	2.21E+00	1.06E−01
CEC06	1.19E+01	5.92E−01	4.42E+00	2.16E+00	1.21E+03	6.00E−01	9.90E+00	1.64E+00	1.07E+01	1.03E+00	6.08E+00	1.49E+00
CEC07	1.24E+03	2.46E+02	4.47E+02	2.70E+02	1.20E+04	1.36E+01	5.79E+02	3.29E+02	4.91E+02	1.95E+02	4.10E+02	2.91E+02
CEC08	6.59E+00	3.27E−01	5.79E+00	4.13E−01	6.10E+00	7.57E−01	6.87E+00	5.02E−01	6.91E+02	4.27E−01	6.37E+00	5.86E−01
CEC09	1.94E+03	4.83E+02	2.74E+00	3.30E−01	2.00E+02	1.59E−10	6.05E+00	2.87E+00	5.94E+02	1.66E+00	3.67E+00	2.36E−01
CEC10	2.05E+01	1.24E−01	2.01E+01	1.41E−01	2.72E+00	8.88E−16	2.13E+01	1.72E−01	2.13E+01	1.11E−01	2.10E+01	7.80E−02

Table 3
P-values of Wilcoxon rank sum test.

Function	BMO	PSO	MFO	DA	SSA	EMA	HBA
F1	1.12E−07	1.81E−06	1.81283E−06	0.0120476	1.86E−09	1.82423E−06	1.82423E−06
F2	1.83E−10	1.86E−09	3.23914E−06	1.86E−09	1.86E−09	1.86E−09	1.86265e−9
F3	1.81E−07	1.86E−09	1.86E−09	1.86E−09	1.86E−08	*0.381798*	1.86E−09
F4	1.86E−09	1.86E−09	1.86E−09	1.86E−09	4.4219E−06	1.86E−09	1.86E−09
F5	1.86265e−9	1.30E−08	*0.271006*	1.30E−08	*0.74566*	1.81055E−06	1.74566E−06
F6	1.86E−09	1.75E−06	1.86E−09	*0.100397*	1.86E−09	1.86E−09	1.81055E−06
F7	*0.612006*	*0.06341*	1.86E−09	1.86E−09	1.86E−09	1.86E−09	1.82423E−06
F8	1.77562E−06	6.04E−05	1.82194E−06	1.86E−09	3.73E−09	1.82194E−06	8.54E−07
F9	1.22E−04	1.30E−08	1.82423E−06	1.86E−09	1.86E−09	*0.417114*	0.047259
F10	1.24E−09	3.05E−05	*0.823577*	1.30E−08	1.86E−09	1.86E−09	1.86E−09
F11	2.59E−08	1.86E−09	1.86E−09	1.30E−08	1.86E−09	3.04952E−05	2.01625E−06
F12	8.63E−09	1.79E−06	1.86E−09	1.746E−06	*0.87121*	1.82309E−06	1.82309E−06
F13	2.61E−08	*0.7691*	1.86E−09	*0.452164*	1.86E−09	1.82423E−06	1.82194E−06
F14	1.14E−02	0.000747	*0.0734822*	1.30E−08	0.0006871	3.04952E−05	1.80715E−06
F15	1.99E−06	1.30E−08	0.000231637	1.30E−08	0.0013406	1.81055E−06	1.82309E−06
F16	1.82423E−06	1.82E−06	1.82423E−06	1.746E−06	1.86E−09	1.86E−09	1.18043E−05
F17	1.81055E−06	1.81E−06	1.81055E−06	1.82E−06	1.8242E−06	1.81966E−06	1.81055E−06
F18	1.86E−09	1.79E−06	1.86E−09	1.86E−09	1.8242E−06	0.000797898	1.86E−09
F19	1.7857E−06	1.86E−09	1.7857E−06	1.86E−09	1.7857E−06	1.86E−09	1.7857E−06
F20	1.86E−09	1.86E−09	1.86E−09	1.30E−08	1.86E−09	1.86E−09	1.86E−09
F21	1.86E−09	*0.36235*	1.86E−09	1.86E−09	1.86E−09	3.73E−09	1.86E−09
F22	1.86E−09	1.09E−09	1.085E−09	1.085E−09	1.86E−09	1.86E−09	1.86E−09
F23	1.86E−09	1.09E−09	1.86E−09	1.86E−09	3.73E−09	1.86E−09	1.86E−09

3.4. Comparison on Wilcoxon-signed rank test

The non-parametric Wilcoxon test is then used to assess GBO's efficacy further. The Wilcoxon-signed rank test is a non-parametric statistic because the best score from a simulation does not come from a normal distribution. In this case, the medians of the two samples were compared on the assumption that they are similar. In this study, we compare GBO against other algorithms using a two-sample, rank-sum test of the hypothesis for each benchmark function. Table 3 displays the performance comparison results between GBO and the other chosen algorithms and associated p-values. The suggested GBO is much better than the chosen algorithms at the 0.05 level, as shown in the table, with the highlighted results being the only for each benchmark function provided by GBO and the chosen algorithms. As GBO is usually statistically significant, we may conclude that the null hypothesis is false. This demonstrates that, for the chosen benchmark functions, GBO achieved very competitive performance.

All the algorithms we examined were simulated in the same framework, on a computer with the same performance and features, so that we could make an accurate comparison.

4. Real application

By using the recommended GBO in the real world and comparing the results to those of the most recent algorithms, its performance and effectiveness have been measured. Time series forecasting of pandemic epidemics like COVID-19, dengue disease, etc. is one such use case. Using the data, we tried to predict the number of confirmed cases per week while taking the cumulative vaccination rate into account. The GBO flowchart depicted in Fig. 8 was followed for the implementation. In this study, a different technique called the least-square support vector machine was used to find the objective function (LSSVM). It is extremely crucial to notice that this has been done in the way that the article specifies [6–8].

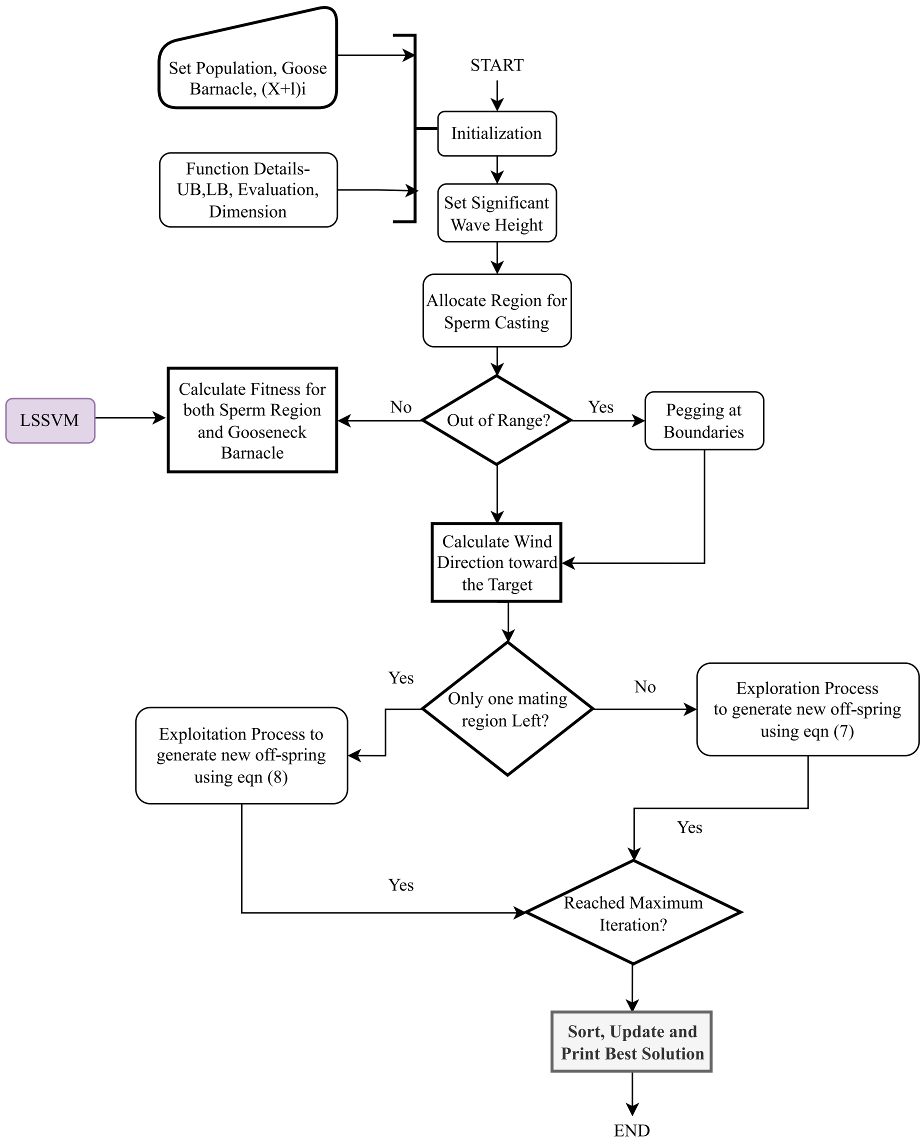

Fig. 8. Flow diagram of GBO.

Even though many recent studies [2,5,9,16,19,28,43,44,64,69,77,84] have used better models and looked at different situations and parameters related to COVID-19 control measures, the implementation used in these studies is better at predicting confirmed cases when the total number of vaccinations is considered. In this setup, GBO optimizes the LSSVM kernel parameter, and LSSVM predicts confirmed cases while taking vaccination into account.

Establishing an X population is the first step in the GBO process flow, from which sperm casting and the integration of a spiral mating zone will produce new offspring. After being evaluated based on factors such as wind speed and food availability, the offspring are then united with their parents. Finally, the best solution available when sorting is to elevate the top of the population.

4.1. Real data pre-processing for Covid-19 confirmed cases prediction with vaccination

Daily data is gathered from January 1, 2020, until October 27, 2022. Then, it will be divided into training, validation, and testing sets. The entire cumulative validated COVID-19 dataset [60], is compiled and kept up to date by combining the daily vaccination case [18] following same date. February 24, 2021, marks the start of a daily collection and gives Malaysia the total number of vaccinations since February 24, 2021. Table 4 illustrates the dataset's schematic concepts, where TCV-W1D1 represents Total Cumulative Vaccination on the first day of the first week, and so on. The dataset has been used to predict the weekly confirmed cases of COVID-19 considering the total vaccination (counting those who have completed the second vaccination dosage).

Table 4
Performance evaluation of different algorithms.

Input							Output
Total cumulative vaccination	Cumulative confirmed cases						Weekly confirmed cases prediction
TCV-W1D1	W1D1	W1D2	W1D3	W1D4	W1D5	W1D6	W1D7
TCV-W2D1	W2D1	W2D2	W2D3	W2D4	W2D5	W2D6	W2D7
TCV-W3D1	W3D1	W3D2	W3D3	W3D4	W3D5	W3D6	W3D7
TCV-W4D1	W4D1	W4D2	W4D3	W4D4	W4D5	W4D6	W4D7
TCV-W5D1	W5D1	W5D2	W5D3	W5D4	W5D5	W5D6	W5D7
TCV-W6D1	W6D1	W6D2	W6D3	W6D4	W6D5	W6D6	W6D7
TCV-W7D1	W7D1	W7D2	W7D3	W7D4	W7D5	W7D6	W7D7

Table 5
Property setting of different algorithms.

Property	Parameter value
Population size	50
Maximum iteration	500
Lower bound	1
Upper bound	1000

4.2. Performance evaluation

Since time series forecasting models have been evaluated through metrics such as, Mean Absolute Percentage Error (MAPE), Theils'U and accuracy, these performance matrices for regression is defined as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} |(y_{predicted} - y_{actual})/y_{actual}| * 100\% \tag{14}$$

$$Theils'U = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_{actual} - y_{predicted})^2}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_{actual})^2} + \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_{predicted}\right)^2}} \tag{15}$$

$$Accuracy = 1 - MAPE \tag{16}$$

Here,
n = The number of test instances,
$y_{predicted}$ = the predicted values at ith time,
y_{actual} = the actual values at ith time.

The above Eqs. (14)–(16) are the common evaluation indicators that define the error rate of the prediction model for regression. Their values should be as small as possible. Theil's U, also known as Theil's inequality coefficient or Theil's entropy, is a statistical measure used to assess the relative effectiveness of different entities or groups within a dataset. In the context of evaluating optimization algorithms, Theil's U quantifies the proportion of improvement achieved by one algorithm over another. A higher Theil's U value indicates that one algorithm's performance is significantly better or worse than another algorithm being compared, providing insights into their relative efficiency and effectiveness in solving optimization problems.

4.3. Properties setting

Each optimized LSSVM model used in the experiment was previously executed with identical property settings (Table 5): GBO-LSSVM, BMO-LSSVM, PSO-LSSVM, SSA-LSSVM, DA-LSSVM, EMA-LSSVM, HBA-LSSVM, and MVO-LSSVM. Dimensions such as population size, maximum iterations, and upper and lower limits for LSSVM hyperparameters are interesting. It is only through trial and error that property prices may be established. The population number is also large enough to achieve the goals.

Like other metaheuristic algorithms, GBO-LSSVM may be used to solve real-world, application-specific problems. This section uses GBO and LSSVM to predict the time series of COVID-19 instances that have already been checked. This document concisely sums up the data gathered from the various analyses. In Table 6, we can see how the prediction performances of all the hybrid algorithms used in this research stack up of verified cases taking the entire vaccination rate into account.

Table 7 compares the actual values of confirmed cases considering the total vaccination to the goal values of all the hybrid algorithms implemented in this research.

Fig. 9 shows the comparison between the target and the predicted values. The confirmed cases prediction with total vaccination in Malaysia using GBO-LSSVM, BMO-LSSVM, and HBA-LSSVM, EMA-LSSVM, DA-LSSVM, MVO-LSSVM, PSO-LSSVM, SSA-LSSVM where GBO-LSSVM outperforms in confirmed cases prediction.

Table 6
Performance evaluation of different algorithms.

Performance comparison			
Algorithms	MAPE	Accuracy	TheilsU
GBO-LSSVM	***0.002211***	***0.997789***	***0.047***
EMA-LSSVM	0.00361	0.9964	0.069
HBA-LSSVM	0.003607	0.996393	0.061
PSO-LSSVM	0.004655	0.9953	0.069
DA-LSSVM	0.0069	0.9931	0.083
BMO-LSSVM	0.011887	0.988113	0.119
SSA-LSSVM	0.0224	0.9776	0.159
MVO-LSSVM	0.02574	0.9743	0.1603

Table 7
Target vs. GBO-LSSVM vs. BMO-LSSVM, HBA-LSSVM, EMA-LSSVM, MVO-LSSVM, PSO-LSVM, SSA-LSSVM.

Week	Target	GBO-LSSVM	BMO-LSSVM	HBA-LSSVM	EMA-LSSVM	MVO-LSSVM	PSO-LSSVM	SSA-LSSVM	DA-LSSVM
67	4 506 510	4 496 595.678	4 452 882.531	4 490 255.018	4 490 286.564	4 390 692.693	4 485 329.403	4 405 564.176	4 475 415.081
68	4 517 447	4 507 508.617	4 463 689.381	4 501 152.569	4 501 184.191	4 401 348.612	4 496 214.999	4 416 256.187	4 486 276.616
69	4 530 312	4 520 345.314	4 476 401.287	4 513 971.165	4 514 002.877	4 413 882.982	4 509 019.534	4 428 833.011	4 499 052.847
70	4 544 626	4 534 627.823	4 490 544.951	4 528 233.534	4 528 265.346	4 427 829.112	4 523 266.258	4 442 826.378	4 513 268.081
71	4 560 583	4 550 549.717	4 506 312.062	4 544 132.977	4 544 164.901	4 443 376.017	4 539 148.26	4458425.941	4529114.977
72	4 578 741	4 568 667.77	4 524 253.982	4 562 225.481	4 562 257.532	4 461 067.356	4 557 220.917	4 476 177.202	4 547 147.687
73	4 600 736	4 590 614.381	4 545 987.242	4 584 141.145	4 584 173.35	4 482 497.085	4 579 112.541	4 497 679.514	4 568 990.922
74	4 629 963	4 619 777.081	4 574 866.44	4 613 262.723	4 613 295.133	4 510 972.951	4 608 202.174	4 526 251.829	4 598 016.255
75	4 659 710	4 649 458.638	4 604 259.451	4 642 902.426	4 642 935.044	4 539 955.453	4 637 809.363	4 555 332.496	4 627 558.001

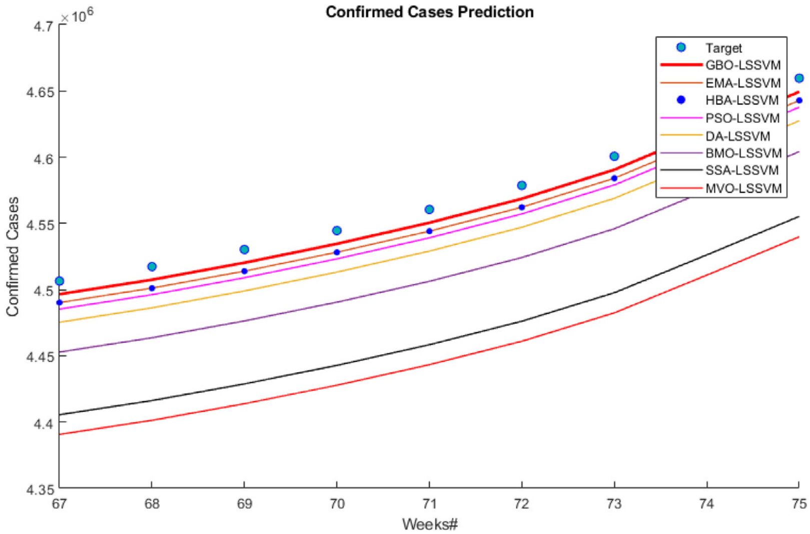

Fig. 9. Prediction comparison between algorithms.

5. Conclusion and future work

The Gooseneck Barnacle Optimization Algorithm (GBO) is a new population-based algorithm inspired by the natural mating behaviour of gooseneck barnacles, which involves sperm casting and self-fertilization. GBO is designed to optimize reproductive success, mimicking the barnacles' mating behaviour. The primary improvement of GBO over the original Barnacle Mating Optimizer (BMO) lies in accurately capturing the unique static and dynamic mating behaviours specific to gooseneck barnacles. Unlike BMO, which provided a generalized representation of barnacle mating by only considering the penis length to generate offspring, GBO models the navigational sperm casting properties, food availability, wind direction, and intertidal zone wave movement experienced by gooseneck barnacles during mating. This level of detail enhances the algorithm's effectiveness in optimization. GBO balances exploration and exploitation, which is crucial for effective optimization. The algorithm leverages the behaviours of gooseneck barnacle groups to execute exploration through sperm casting and exploitation through self-fertilization. This two-phase process enhances the algorithm's search capabilities and improves convergence to the global optimum. A thorough study of GBO's performance was done using a variety of benchmark functions, the CEC'19 test suite, and real-world situations. Statistical analyses, exploration-exploitation ratios, and convergence curves were utilized to measure the algorithm's effectiveness in searching for optimal solutions. In addition to evaluating GBO's performance, a comparative analysis was conducted against several other optimization algorithms on the same experimental platform, including GA, PSO, DA, EMA, HBA, BMO, MVO, and SSA. The results demonstrated that GBO outperformed these algorithms, showcasing its superior convergence speed and ability to balance exploration and exploitation.

Based on the findings, future research on GBO will focus on further enhancements. Introducing features such as chaotic maps, binary objectives, and multi-objectives will enable GBO to tackle more complex and large-scale optimization problems effectively. Additionally, food attractiveness can be added as a constraint during reproduction. Furthermore, future research endeavours may include the CEC2020 and CEC2021 datasets to increase the breadth of our investigation and better illustrate the benefits and limitations of the GBO. This will enable us to evaluate the efficacy of the GBO on application-specific functions and compare it to other cutting-edge algorithms for multi-objective optimization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research study was supported by Ministry of Education Malaysia (MOE) and Universiti Malaysia Pahang under Fundamental Research Grant Scheme (FRGS/1/2019/ICT02/UMP/03/1) & (#RDU1901133).

References

[1] J.M. Abdullah, T. Ahmed, Fitness dependent optimizer: inspired by the bee swarming reproductive process, IEEE Access 7 (2019) 43473–43486.

[2] E. Acosta-González, J. Andrada-Félix, F. Fernández-Rodríguez, On the evolution of the COVID-19 epidemiological parameters using only the series of deceased. A study of the spanish outbreak using Genetic Algorithms, Math. Comput. Simulation 197 (2022) 91–104.

[3] ADW: Lepas anatifera: Information, 2023, https://animaldiversity.org/site/accounts/information/Lepas_anatifera.html, Accessed on: 3/19/2023.

[4] J.O. Agushaka, A.E. Ezugwu, L. Abualigah, Gazelle optimization algorithm: a novel nature-inspired metaheuristic optimizer, Neural Comput. Appl. 35 (2023) 4099–4131, http://dx.doi.org/10.1007/s00521-022-07854-6.

[5] M. Ahmed, A.J. Mohamad, M. Rahman, M.H. Sulaiman, M.A. Kashem, A novel hybrid evolutionary mating algorithm for covid19 confirmed cases prediction based on vaccination, in: 2023 International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM), IEEE, 2023, pp. 1–6.

[6] M. Ahmed, M.H. Sulaiman, A.J. Mohamad, Improved barnacle mating optimizer-based least square support vector machine to predict COVID-19 confirmed cases with total vaccination, Cybern. Inf. Technol. 23 (1) (2023) 125–140.

[7] M. Ahmed, M.H. Sulaiman, A.J. Mohamad, M. Rahman, A hybrid method for analyzing the situation based on cumulative fully vaccinated and confirmed cases of Covid-19 in Malaysia, in: 2022 4th International Conference on Sustainable Technologies for Industry 4.0 (STI), IEEE, 2022, pp. 1–6.

[8] M. Ahmed, M.H. Sulaiman, A.J. Mohamad, M. Rahman, An improved optimization algorithm-based prediction approach for the weekly trend of COVID-19 considering the total vaccination in Malaysia: A novel hybrid machine learning approach, in: Proceedings of the Fourth International Conference on Trends in Computational and Cognitive Engineering: TCCE 2022, Springer Nature Singapore, Singapore, 2023, pp. 209–223.

[9] M. Anand, P. Danumjaya, P.R.S. Rao, A nonlinear mathematical model on the Covid-19 transmission pattern among diabetic and non-diabetic population, Math. Comput. Simulation (2023).

[10] M. Azizi, S. Talatahari, A.H. Gandomi, Fire Hawk Optimizer: a novel metaheuristic algorithm, Artif. Intell. Rev. 56 (2023) 287–363, http://dx.doi.org/10.1007/s10462-022-10173-w.

[11] M. Barazandeh, C.S. Davis, C.J. Neufeld, D.W. Coltman, A.R. Palmer, Something Darwin didn't know about barnacles: spermcast mating in a common stalked species, Proc. R. Soc. B: Biol. Sci. 280 (1754) (2013) 20122919.

[12] M. Barazandeh, C.S. Davis, C.J. Neufeld, D.W. Coltman, A.R. Palmer, Something Darwin didn't know about barnacles: spermcast mating in a common stalked species, Proc. R. Soc. B 280 (1754) (2013) 20122919, http://dx.doi.org/10.1098/rspb.2012.2919.

[13] Angel Borja, Pedro Liria, Iñigo Muxika, Juan Bald, Relationships between wave exposure and biomass of the goose barnacle (Pollicipes pollicipes, Gmelin, 1790. in the Gaztelugatxe Marine Reserve (Basque Country, northern Spain), ICES J. Mar. Sci. 63 (2006) 626, http://dx.doi.org/10.1016/j.icesjms.2005.12.008.

[14] M. Braik, M.H. Ryalat, H. Al-Zoubi, A novel meta-heuristic algorithm for solving numerical optimization problems: Ali Baba and the forty thieves, Neural Comput. Appl. 34 (1) (2022) 409–455.

[15] T. Burel, G. Schaal, J. Grall, M. Le Duff, E. Ar Gall, Clear-cut wave height thresholds reveal dominance shifts in assemblage patterns on rocky shores, Mar. Ecol. Prog. Ser. 683 (2022) 21–36, http://dx.doi.org/10.3354/meps13945.

[16] Z. Chen, L. Feng, H.A. Lay Jr., K. Furati, A. Khaliq, SEIR model with unreported infected population and dynamic parameters for the spread of COVID-19, Math. Comput. Simul. 198 (2022) 31–46.

[17] CA. Coello Coello, Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art, Comput. Methods Appl. Mech. Eng. 191 (2002) 1245–1287.

[18] Coronavirus (COVID-19) vaccinations - our world in data, 2023, https://ourworldindata.org/covid-vaccinations, Accessed on: 3/19/2023.

[19] P. Cumsille, Ó. Rojas-Díaz, P.M. de Espanés, P. Verdugo-Hernández, Forecasting COVID-19 Chile'second outbreak by a generalized SIR model with constant time delays and a fitted positivity rate, Math. Comput. Simulation 193 (2022) 1–18.

[20] M. Dehghani, Z. Montazeri, E. Trojovská, P. Trojovský, Coati optimization algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems, Knowl.-Based Syst. 259 (2023) 110011.

[21] M. Dorigo, M. Birattari, T. Stutzle, Ant colony optimization, Comput. Intell. Mag. IEEE 1 (4) (2006) 28–39.

[22] A.K. Dutta, R.U. Mageswari, A. Gayathri, J.M. Dallfin Bruxella, M.K. Ishak, S.M. Mostafa, H. Hamam, Barnacles mating optimizer with deep transfer learning enabled biomedical malaria parasite detection and classification, Comput. Intell. Neurosci. (2022).

[23] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 1995, pp. 39–43, http://dx.doi.org/10.1109/MHS.1995.494215.

[24] M.Abd. Elaziz, A.A. Ewees, M.A. Al-qaness, L. Abualigah, R.A. Ibrahim, Sine–Cosine-Barnacles Algorithm Optimizer with disruption operator for global optimization and automatic data clustering, Expert Syst. Appl. 207 (2022) 117993.

[25] N. Eslami, S. Yazdani, M. Mirzaei, E. Hadavandi, Aphid–Ant Mutualism: A novel nature-inspired metaheuristic algorithm for solving optimization problems, Math. Comput. Simulation 201 (2022) 362–395.
[26] G. Fan, M. Li, X. Chen, X. Dong, K. Jermsittiparsert, Analysis of a multi-objective hybrid system to generate power in different environmental conditions based on improved the Barnacles Mating Optimizer Algorithm, Energy Rep. 7 (2021) 2950–2961.
[27] J.N. Fernandes, D. Jacinto, N. Penteado, A. Sousa, D. Mateus, M.I. Seabra …, T. Cruz, Ten years of monitoring recruitment of the edible stalked barnacle Pollicipes pollicipes: linking to oceanographic variability, Limnol. Oceanogr. 66 (6) (2021) 2309–2318.
[28] J.K. Ghosh, S.K. Biswas, S. Sarkar, U. Ghosh, Mathematical modelling of COVID-19. a case study of Italy, Math. Comput. Simulation 194 (2022) 1–18.
[29] DE. Goldberg, Genetic algorithms in search, optimization, and machine learning, 1989.
[30] F.A. Hashim, E.H. Houssein, K. Hussain, M.S. Mabrouk, W. Al-Atabany, Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems, Math. Comput. Simulation 192 (2022) 84–110.
[31] RL. Haupt, SE. Haupt, Practical Genetic Algorithms, 2nd ed., Wiley, 2004.
[32] J.M. Hoch, Variation in penis morphology and mating ability in the acorn barnacle, Semibalanus balanoides, J. Exp. Mar. Biol. Ecol. 359 (2) (2008) 126–130.
[33] J.M. Hoch, Variation in penis morphology and mating ability in the acorn barnacle, Semibalanus balanoides, J. Exp. Mar. Biol. Ecol. 359 (2) (2008) 126–130, http://dx.doi.org/10.1016/j.jembe.2008.03.002.
[34] J.M. Hoch, Adaptive plasticity of the penis in a simultaneous hermaphrodite, Evolution 63 (8) (2009) 1946–1953, http://dx.doi.org/10.1111/j.1558-5646.2009.00668.x.
[35] J.M. Hoch, Effects of crowding and wave exposure on penis morphology of the acorn barnacle, Semibalanus balanoides, Mar. Biol. 157 (12) (2010) 2783–2789, http://dx.doi.org/10.1007/s00227-010-1536-z.
[36] JH. Holl, Genetic algorithms, Sci. Am. 267 (1992) 66–72.
[37] E.H. Houssein, M.R. Saad, F.A. Hashim, H. Shaban, M. Hassaballah, Lévy flight distribution: A new metaheuristic algorithm for solving engineering optimization problems, Eng. Appl. Artif. Intell. 94 (2020) 103731.
[38] Intertidal zone: videos, info and lessons from untamed science, 2023, https://untamedscience.com/biology/biomes/intertidal-zone-aquatic-biome/, Accessed on: 3/19/2023.
[39] S.S. Jadon, J.C. Bansal, R. Tiwari, H. Sharma, Accelerating Artificial Bee Colony algorithm with adaptive local search, Memet. Comput. (2015) 1–16.
[40] H. Jia, K. Sun, Improved barnacles mating optimizer algorithm for feature selection and support vector machine optimization, Pattern Anal. Appl. 24 (3) (2021) 1249–1274.
[41] T. Kawai, M. Tokeshi, Variable modes of facilitation in the upper intertidal: goose barnacles and mussels, Mar. Ecol. Prog. Ser. 272 (2004) 203–213.
[42] L. Khandeparker, A.C. Anil, Underwater adhesion: the barnacle way, Int. J. Adhes. Adhes. 27 (2) (2007) 165–172.
[43] O. Koutou, B. Sangaré, Mathematical analysis of the impact of the media coverage in mitigating the outbreak of COVID-19, Math. Comput. Simulation 205 (2023) 600–618.
[44] R.P. Kumar, P.K. Santra, G.S. Mahapatra, Global stability and analysing the sensitivity of parameters of a multiple-susceptible population model of SARS-CoV-2 emphasising vaccination drive, Math. Comput. Simulation 203 (2023) 741–766.
[45] LB. Ladah, FJ. Tapia, Internal wave intensity and angle of propagation modulate small-scale settlement patterns of intertidal barnacles during peak recruitment, Mar. Ecol. Prog. Ser. 678 (2021) 81–93, http://dx.doi.org/10.3354/meps13876.
[46] H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, Appl. Soft Comput. 10 (2010) 629–640.
[47] RT. Marler, JS. Arora, Survey of multi-objective optimization methods for engi-neering, Struct. Multidiscip. Optim. 26 (2004) 369–395.
[48] S.H.H. Mehne, S. Mirjalili, Moth-flame optimization algorithm: theory, literature review, and application in optimal nonlinear feedback control design, Nat.-Inspired Optim.: Theor. Lit. Rev. Appl. (2020) 143–166.
[49] B. Merikhi, MR. Soleymani, Automatic data clustering framework using nature-inspired binary optimization algorithms, IEEE Access 9 (2021) 93703–93722, http://dx.doi.org/10.1109/ACCESS.2021.3091397.
[50] S. Mirjalili, The ant lion optimizer, Adv. Eng. Softw. 83 (2015) 80–98.
[51] S. Mirjalili, Dragon_y algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, Neural Comput. Appl. 27 (4) (2015) 1053_1073.
[52] A. Mirjalili, S. Mirjalili, Seyedali mirjalili, 2015, [Online]. Available: http://www.alimirjalili.com/Projects.html, Accessed: April. 01, 2023.
[53] S. Mirjalili, SM. Mirjalili, A. Hatamlou, Multi-Verse Optimizer: a nature-inspired algorithm for global optimization, Neural Comput. Appl. (2015) http://dx.doi.org/10.1007/s00521-015-1870-7.
[54] S. Mirjalili, SM. Mirjalili, A. Lewis, Grey Wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61, http://dx.doi.org/10.1016/j.advengsoft.2013.12.007.
[55] Z. Mustaffa, M.H. Sulaiman, COVID-19 confirmed cases prediction in China based on barnacles mating optimizer-least squares support vector machines, Cybern. Inf. Technol. 21 (4) (2021) 62–76.
[56] Z. Mustaffa, M.H. Sulaiman, Stock price predictive analysis: An application of hybrid barnacles mating optimizer with artificial neural network, Int. J. Cogn. Comput. Eng. (2023).
[57] Z. Mustaffa, M.H. Sulaiman, M.F.M. Mohsin, Y. Yusof, F. Ernawan, B. Yusob, N.M. Noor, An application of barnacle mating optimizer in infectious disease prediction: A dengue outbreak cases, Iraqi J. Sci. 213 (2020) 2–2141.
[58] N. Nabipour, M. Dehghani, A. Mosavi, S. Shamshirb, Short-term hydrological drought forecasting based on different nature-inspired optimization algorithms hybridized with artificial neural networks, IEEE Access 8 (2020) 15210–15222, http://dx.doi.org/10.1109/ACCESS.2020.2964584.
[59] C.J. Neufeld, A.R. Palmer, Precisely proportioned: intertidal barnacles alter penis form to suit coastal wave action, Proc. R. Soc. B: Biol. Sci. 275 (1638) (2008) 1081–1087, http://dx.doi.org/10.1098/rspb.2007.1760.
[60] Novel coronavirus (COVID-19) cases data - humanitarian data exchange, 2023, https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases, Accessed on: 3/19/2023.
[61] H. Ouyang, J. Chen, S. Li, J. Xiang, Z.H. Zhan, Altruistic population algorithm: A metaheuristic search algorithm for solving multimodal multi-objective optimization problems, Math. Comput. Simulation (2023).
[62] J.S. Pan, L.G. Zhang, R.B. Wang, V. Snášel, S.C. Chu, Gannet optimization algorithm: A new metaheuristic algorithm for solving engineering optimization problems, Math. Comput. Simulation 202 (2022) 343–373.
[63] Pictured: Goose barnacles potentially worth thousands wash up on english coast, 2023, https://www.telegraph.co.uk/news/2023/01/10/pictured-goose-barnacles-potentially-worth-thousands-wash-english/, Accessed on: 3/19/2023.
[64] O. Postavaru, S.R. Anton, A. Toma, COVID-19 pandemic and chaos theory, Math. Comput. Simul. 181 (2021) 138–149.
[65] K.V. Price, N.H. Awad, M.Z. Ali, P.N. Suganthan, The 100-Digit Challenge: Problem de_nitions and Evaluation Criteria for the 100-Digit Challenge Special Session and Competition on Single Objective Numerical Optimization, Tech. Rep., School Elect. Electron. Eng. Nanyang Technol. Univ. Singapore, 2018.
[66] T. Ray, P. Saini, Engineering design optimization using a swarm with an intelli- gent information sharing among individuals, Eng. Optim. 33 (2001) 735–748.
[67] Researchers discover new method of barnacle sex | science| smithsonian magazine, 2023, https://www.smithsonianmag.com/science-nature/researchers-discover-new-method-of-barnacle-sex-2165681/, Accessed on: 3/19/2023.

[68] RM. Rizk-Allah, AE. Hassanien, M. Elhoseny, M. Gunasekaran, A new binary salp swarm algorithm: development and application for optimization tasks, Neural Comput. Appl. 31 (5) (2019) 1641–1663, http://dx.doi.org/10.1007/s00521-018-3613-z.
[69] S. Saha, G. Samanta, J.J. Nieto, Impact of optimal vaccination and social distancing on COVID-19 pandemic, Math. Comput. Simulation 200 (2022) 285–314.
[70] Significant wave height, significant wave height is defined, followed by a random wave analysis. From: Free-surface flow:, 2019, https://www.sciencedirect.com/topics/engineering/significant-wave-height, Accessed on: 7/19/2023.
[71] Significant Wave Height, Finnish Meteorological Institute, 2023, https://en.ilmatieteenlaitos.fi/significant-wave-height, Accessed on: 7/19/2023.
[72] Significant Wave Height, Wattnotions, 2023, https://www.wattnotions.com/openwave-8-calculating-significant-wave-height/, Accessed on: 7/19/2023.
[73] Significant Wave Height, Wikipedia, 2023, https://en.wikipedia.org/wiki/Significant_wave_height, Accessed on: 7/19/2023.
[74] G. Singh, A. Singh, Extension of particle swarm optimization algorithm for solving transportation problem in fuzzy environment, Appl. Soft Comput. 110 (2021) 107619, http://dx.doi.org/10.1016/j.asoc.2021.107619.
[75] R. Storn, K. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, J. Glob. Optim. 11 (4) (1997) 341–359, http://dx.doi.org/10.1023/A:1008202821328.
[76] M.H. Sulaiman, Z. Mustaffa, M.M. Saari, H. Daniyal, Barnacles mating optimizer: A new bio-inspired algorithm for solving engineering optimization problems, Eng. Appl. Artif. Intell. 87 (2020) 103330.
[77] J. Waku, K. Oshinubi, J. Demongeot, Maximal reproduction number estimation and identification of transmission rate from the first inflection point of new infectious cases waves: COVID-19 outbreak example, Math. Comput. Simulation 198 (2022) 47–64.
[78] G-G. Wang, AH. Gandomi, X. Zhao, HCE. Chu, Hybridizing harmony search algorithm with cuckoo search for global numerical optimization, Soft Comput. (2014) http://dx.doi.org/10.1007/s00500-014-1502-7.
[79] G. Wang, L. Guo, H. Wang, H. Duan, L. Liu, J. Li, Incorporating mutation scheme into krill herd algorithm for global numerical optimization, Neural Comput. Appl. 24 (2014) 853–871.
[80] G-G. Wang, M. Lu, Y-Q. Dong, X-J. Zhao, Self-adaptive extreme learning machine, Neural Comput. Appl. (2015) http://dx.doi.org/10.1007/s00521-015-1874-3.
[81] H. Wang, Z. Wu, S. Rahnamayan, H. Sun, Y. Liu, J.-s. Pan, Multi-strategy ensemble artificial bee colony algorithm, Inform. Sci. 279 (2014) 587–603.
[82] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1997) 67–82.
[83] Z. Yang, Q. Liu, L. Zhang, J. Dai, N. Razmjooy, Model parameter estimation of the PEMFCs using improved barnacles mating optimization algorithm, Energy 212 (2020) 118738.
[84] P. Yarsky, Using a genetic algorithm to fit parameters of a COVID-19 SEIR model for US states, Math. Comput. Simulation 185 (2021) 687–695.
[85] M. Zhang, W. Luo, X. Wang, Differential evolution with dynamic stochastic se- lection for constrained optimization, Inf. Sci. 178 (2008) 3043–3074.